

# CBCAN

---

## SDO 及 NMT 服務的 PLC 暫存器界面

V1.2

15 June 2012

## 目錄

|  |          |
|--|----------|
| 目錄.....                                    | 2        |
| 表目錄.....                                   | 3        |
| <b>1. 簡介</b> .....                         | <b>5</b> |
| <b>2. SDO 服務</b> .....                     | <b>5</b> |
| <b>3. NMT 服務</b> .....                     | <b>6</b> |
| <b>4. 範例 – NMT 服務 (Start Remote)</b> ..... | <b>7</b> |

## 表目錄

|   |   |
|---|---|
| 表 1: PLC Registers for SDO Service .....                | 5 |
| 表 2: Registers Status after SDO Service Execution ..... | 5 |
| 表 3: SDO Error Code.....                                | 6 |

## 圖目錄

|  |   |
|--|---|
| 圖 1： NMT Start Remote command – 階梯程式範例 ..... | 8 |
|--|---|

## CBCAN SDO 和 NMT 服務的 PLC 暫存器界面

### 1. 簡介

目前的 Winproladder 設計並沒有提供 SDO 和 NMT 服務的指令，因此在某些特定應用必需使用 SDO 或 NMT 就必須根據此文件提供的 PLC 暫存器界面來設計 Ladder 程式。

### 2. SDO 服務

當 CBCAN 模組在 FBs PLC 使用時，PLC 會經由暫存器 R3700~R3769 來控制 CBCAN 模組。完成一個 SDO 服務的主要資訊包含有 SDO 從站的 node ID (R3702)、從站物件字典的 index 及 sub-index (R3703 and R3704)、要存取的資料大小(R3705)以及資料本身 (R3706~R3769); 在上述資料都準備妥當且寫到相對應的暫存器了之後，就可以將“CA0Eh”寫入到暫存器 R3700 來通知 CBCAN 模組執行 SDO 服務。

而執行結果會在指令執行結束後顯示在同樣範圍的暫存器 R3700~R3769，請參閱下方的表格來進一步了解 SDO 服務的詳細使用方法。

**表 1: PLC Registers for SDO Service**

| 序號 | 暫存器   | 功能   |
|----|-------|--|
| 1  | R3700 | Execute Flag = CA0EH   |
| 2  | R3701 | Command.<br>SDO Read=5055<br>SDO Write = 5066  |
| 3  | R3702 | Node ID( station number, 0~127. If =0, this node)                                    |
| 4  | R3703 | Object Index (0~65535)   |
| 5  | R3704 | Object sub-index (0~255)   |
| 6  | R3705 | Object data size: 1 ~ 128 Byte. Data size to Write or max. data buffer size for read |
| 7  | R3706 | Data word #0   |
| 8  | R3707 | Data word #1   |
| .  | .     | .  |
| 70 | R3769 | Data word #63  |

**表 2: Registers Status after SDO Service Execution**

| 序號 | 暫存器   | 功能            |       |
|----|-------|---------------|-------|
|    |       | Read          | Write |
| 1  | R3700 | Result Code*1 |       |
| 2  | R3701 | No Change     |       |
| 3  | R3702 |               |       |

|    |       |                            |                         |
|----|-------|----------------------------|-------------------------|
| 4  | R3703 |                            |                         |
| 5  | R3704 |                            |                         |
| 6  | R3705 | Actual read data size      | No Change               |
| 7  | R3706 | Data word #0* <sub>2</sub> | No Change* <sub>2</sub> |
| 8  | R3707 | Data word #1* <sub>3</sub> | No Change* <sub>3</sub> |
| .  | .     | .                          | No Change               |
| 70 | R3769 | Data word #63              | No Change               |

\*<sub>1</sub>: Result Code: OK = 0, CMD\_CODE\_ERR = 1, NODE\_ID\_ERR = 2, SDO\_EXEC\_ERR = 4.

\*<sub>2</sub>: 當執行失敗時的SDO error code low word (Result Code= 4)

\*<sub>3</sub>: 當執行失敗時的SDO error code high word (Result Code= 4)

表 3: SDO Error Code

| Error Code Name   | Error Value | Description                          |
|-------------------|-------------|--------------------------------------|
| ABORT_TIME_OUT    | 0x05040000L | SDO service Time out                 |
| ABORT_NO_OBJ      | 0x06020000L | No such object                       |
| ABORT_RO          | 0x06010002L | Attempt to write a read-only object  |
| ABORT_SYS_LENGTH  | 0x06040047L | Data length exceed system allow      |
| ABORT_NO_SEGEMNT  | 0x06010000L | Not support segment transfer         |
| ABORT_OBJ_LENGTH  | 0x06070010L | Not match object length              |
| ABORT_SYNC        | 0x05040001L | Command specifier not valid          |
| ABORT_TOGGLE_BIT  | 0x05030000L | Toggle bit not alternated            |
| ABORT_PARM_LENGTH | 0x06070012L | Length of service parameter too high |
| ABORT_WO          | 0x06010001L | Attempt to read a write-only object  |
| ABORT_READ_LENGTH | 0x05040005L | Object length too big to read        |

### 3. NMT 服務

NMT 服務同樣的也是使用 and SDO 服務一樣的方式來完成，不同點在於使用 NMT 服務時是對特定的物件字典位址做存取“0x1F82” (write service)以及“0x300C” (read service)。

#### NMT Management 服務 – 發送 NMT 服務指令到 NMT 從站

SDO write (R3701): With Object index (R3703) = 0x1f82

| R3704: Sub-Index n | Description                         |
|--------------------|-------------------------------------|
| n = 1 ~ 127        | Node with node ID n to be requested |

|         |                   |
|---------|-------------------|
| n = 128 | Request all nodes |
|---------|-------------------|

R3702: Do not care.

| R3706: Command Code | NMT Service Description     |
|---------------------|-----------------------------|
| 5 or 1              | Start Remote Node           |
| 4 or 2              | Stop Remote Node            |
| 127 or 128          | Enter Pre-Operational State |
| 6 or 129            | Reset Node                  |
| 7 or 130            | Reset Communication         |

從站 **NMT 狀態服務** – 讀取特定從站的 **NMT 狀態**

SDO read (R3701): With Object index (R3703) = 0x300c

Sub-index (R3704) = 0

Node ID (R3702): Requested node ID

Execution Result (R3700 and R3706):

| 暫存器                                | 值     | 描述              |
|------------------------------------|-------|-----------------|
| R3700                              | 0     | OK              |
|                                    | Not 0 | Not OK          |
| R3706 (Only valid when R3700 is 0) | 4     | Stopped         |
|                                    | 127   | Pre-operational |
|                                    | 5     | Operational     |
|                                    | 6     | Not detected    |

#### 4. 範例 – NMT 服務 (Start Remote)

下方階梯程式提供了如何使用 **NMT 服務** 來致能網路上的遠端節點的範例；根據前章節的描述，將各暫存器設定如下表所示便可送出 **NMT** 的命令。

| 暫存器   | 值     | 描述                |
|-------|-------|-------------------|
| R3701 | 5066  | Write command     |
| R3703 | 1F82H | NMT service index |
| R3704 | 128   | Request all nodes |
| R3706 | 5     | Start Remote Node |

在完成上述的暫存器後，最後需將 R3700 填入 CA0Eh 以觸發 NMT 命令。下方範例的第一個 function “17.CMP” 只是來作為 NMT 服務的觸發，使用 NMT 服務並不是必要步驟。

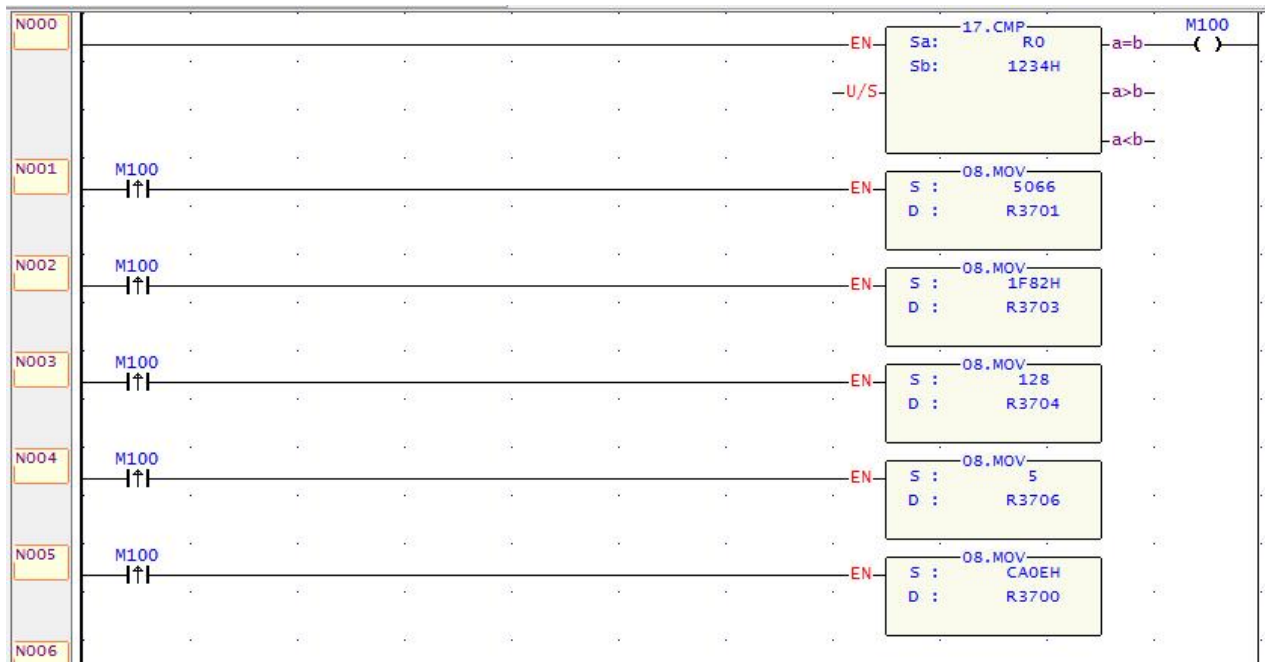


圖 1： NMT Start Remote command – 階梯程式範例